In re application of: Rubin

Serial No.: 10/037,199

Filed on: 01/04/02

For: **OBJECT ORIENTED INFORMATION RETRIEVAL FRAMEWORK MECHANISM**

Mail Stop APPEAL BRIEF - PATENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Transmitted herewith for filing is an **Appeal Brief** in triplicate for the above-identified Application.

[X] Please deduct $500.00 from Deposit Account No. 09-0465 for IBM Corporation to cover the fee under 37 C.F.R. §1.17(f) for the filing of the enclosed appeal brief.

[X] The Commissioner is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account No. 09-0465 for IBM Corporation. A duplicate copy of this sheet is enclosed.

  [X] Any additional filing fees required under 37 C.F.R. §1.16.

  [X] Any patent application processing fees under 37 C.F.R. §1.17.

MARTIN & ASSOCIATES, L.L.C.
P.O. Box 548
Carthage, MO 64836-0548
(417) 358-4700
FAX (417) 358-5757

Respectfully submitted,

By_____
Derek P. Martin
Reg. No. 36,595

I HEREBY CERTIFY THAT THE CORRESPONDENCE TO WHICH THIS STATEMENT IS AFFIXED IS BEING DEPOSITED WITH THE UNITED STATES POSTAL SERVICE, POSTAGE PAID, AS FIRST CLASS MAIL IN AN ENVELOPE ADDRESSED TO: MAIL STOP APPEAL BRIEF - PATENT, COMMISSIONER FOR PATENTS, P.O. BOX 1450, ALEXANDRIA, VA 22313-1450.

Date: January 19, 2006   By: _____

In re application of:	Rubin	Docket No.:	ROC919950062US3
Serial No.:	10/037,199	Group Art Unit:	2191
Filed:	01/04/02	Examiner:	Zhen, Wei Y.

For: OBJECT ORIENTED INFORMATION RETRIEVAL FRAMEWORK MECHANISM

APPEAL BRIEF

Mail Stop APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir/Madam:

This appeal is taken from the Examiner's final rejection, set forth in the Office Action dated 08/22/05, of applicants' claims 1-23. Applicants' Notice of Appeal under 37 C.F.R. § 1.191 was mailed on 11/21/05.

REAL PARTY IN INTEREST

International Business Machines Corporation is the Real Party in Interest.

RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences for this patent application or its parent.

STATUS OF CLAIMS

As filed, this case included claims 1-23. No claim has been amended, and no claim has been cancelled. In the pending office action, claims 1-23 were rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 6,081,798 to Johnson *et al.* (hereinafter "Johnson"). The claims remaining in the case are claims 1-23 as originally filed, all of which stand finally rejected. No claim has been allowed.

STATUS OF AMENDMENTS

No amendments have been filed. Therefore, the claims at issue in this appeal are claims 1-23 as originally filed.

SUMMARY OF CLAIMED SUBJECT MATTER

Independent claim 1 recites:

1. An apparatus comprising:
    at least one processor;
    a memory coupled to the at least one processor;
    a user-extensible object oriented framework residing in the
memory, the framework including at least one core function that cannot be
modified by a user and at least one extensible function defined by a user to
customize the framework and thereby define a desired information
retrieval system, the framework including:
        a load document processor that loads and preprocesses a
    plurality of documents;
        an index processor that creates at least one word index
    corresponding to the plurality of documents; and
        a query processor that receives a query and determines if
    any of the plurality of documents match the query by processing
    the query and comparing the processed query to the plurality of
    words in the at least one word index, thereby providing a query
    result.

The apparatus in claim 1 is shown as 30 in FIG. 8, with the CPU 32 reading on the at least one processor and the main memory 38 reading on the memory coupled to the at least one processor. FIG. 1 is described in the specification at p. 22 line 7 to p. 23 line 11. The user-extensible object oriented framework mechanism in claim 1 is shown in the category diagram of FIG. 14. The load document processor, index processor and query processor in claim 1 are shown as categories load document, build index, and query index in FIG. 14, and as corresponding classes of the framework in FIG. 15. FIGS. 14 and 15 are described in the specification at p. 33 line 14 to p. 35 line 2.

Independent claim 8 recites a program product with limitations similar to claim 1. The user-extensible object oriented framework mechanism in claim 8 is shown in the category diagram of FIG. 14. The load document processor, index processor and query processor in claim 1 are shown as categories load document, build index, and query index in FIG. 14, and as corresponding classes of the framework in FIG. 15. FIGS. 14 and 15 are described in the specification at p. 33 line 14 to p. 35 line 2.

Independent claim 17 recites a method of retrieving information from a plurality of documents comprising steps (1) through (3)(C). Step (1) is shown in the category diagram of FIG. 14, which is discussed in the specification at p. 33 line 14 to p. 34 line 4. Step (2) of extending the object oriented framework mechanism is shown by the example in FIGS. 15-19, which is discussed in the specification at p. 34 line 12 to p. 42 line 2. Step (3) of executing the object oriented framework mechanism is shown in the scenario diagrams of FIGS. 20-23B, which is described in the specification at p. 42 line 3 to p. 50 line 8. Step (3)(A) is shown at 114 in FIG. 9, and in FIG. 10. Step (3)(B) is shown at 116 in FIG. 9, and in FIG. 11. Step (3)(C) is shown at 118 in FIG. 9, and in FIG. 12. FIGS. 9-12 are described in the specification at p. p. 23 line 20 to p. 31 line 12.

3

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The following single ground of rejection is presented for review on this Appeal:

1.   **Whether claims 1-23 are unpatentable as being anticipated under 35 U.S.C. §102(e) by Johnson**

ARGUMENT

Issue 1:   Whether claims 1-23 are unpatentable as being anticipated under 35 U.S.C. §102(e) by Johnson

<u>Claims 1, 8-10 and 17</u>

In the rejection of claim 1, the examiner states that Johnson teaches a load document processor that loads and preprocesses a plurality of documents at col. 2:56 to col. 3:13, col. 7:2-31 and col. 63:63 to col. 64:10. The examiner also states that Johnson teaches an index processor that creates at least one word index corresponding to the plurality of documents at col. 64:29-34 "index definition class of objects" and col. 51:30-65 "IndexDefinition" objects and col. 16:50-67. The examiner then states that Johnson teaches a query processor (E.g., see col. 2:56 to col. 3:13, "query processing") that receives a query and determines if any of the plurality of documents match the query by processing the query and comparing the processed query to the plurality of words in the at least one word index, thereby providing a query result (E.g., see col. 64:34-43). Applicants respectfully assert that the examiner has not identified a teaching in Johnson that reads on each and every limitation in claim 1. As a result, the examiner has failed to establish a prima facie case of anticipation for claim 1 under 35 U.S.C. §102(e).

Claim 1 recites, among other things, a load document processor, a plurality of documents, an index processor, at least one word index corresponding to the plurality of documents, and a plurality of words in the at least one word index. The claimed invention is concerned with documents. The examiner has not specifically identified in Johnson what teaching or feature allegedly reads on the load document processor in the claims. The examiner has not specifically identified in Johnson what teaching or feature allegedly reads on the plurality of documents in the claims. The examiner has not specifically identified in Johnson what teaching or feature allegedly reads on the plurality of words in the at least one word index. Instead, the examiner cites an entire clause in claim 1, then cites several portions of Johnson as allegedly teaching all of the limitations without specifically mapping the teachings of Johnson on the many claim limitations in each clause. If Johnson related to document processing and word indexes that contain a plurality of words, such a simplistic approach might establish a prima facie case of anticipation. However, because Johnson does not teach document processing and word indexes that contain a plurality of words, the examiner's failure to map the teachings of Johnson on the load document processor, documents and plurality of words in claim 1 results in a failure to establish a prima facie case of anticipation for claim 1 under 35 U.S.C. §102(e).

Without a detailed mapping of the teachings of Johnson on the limitations in claim 1, applicant is left to guess what features of Johnson read on what features in the claims. For example, the cited portions of Johnson that allegedly read on the load document processor do not reference documents at all. The most reasonable guess of the examiner's logic is that the examiner believes the case instance descriptions of Johnson read on the documents in claim 1. While Johnson teaches a framework mechanism that works with case instance descriptions, Johnson does not teach "a load document processor that loads and preprocesses a plurality of documents" as recited in claim 1. Even if the case instance descriptions in Johnson can be broadly read on the plurality of

documents in claim 1, nothing in Johnson "loads and preprocesses" the case instance descriptions. For this reason alone, claim 1 is allowable over Johnson.

In rejecting the index processor in claim 1, the examiner cites to the index definition class of objects in Johnson. Johnson clearly teaches an index to the case instance objects at the last line of claim 7 at col. 64, line 33. Note, however, that the index processor in claim 1 "creates at least one word index corresponding to the plurality of documents." Nowhere does Johnson teach or suggest that the index is a word index. For this reason alone, claim 1 is allowable over Johnson.

In rejecting the query processor in claim 1, the examiner cites to the query processing of Johnson at col. 2:56 to col. 3:13. While this cited portion of Johnson does recite query processing, the details of how the query processing is performed are not given. In claim 1, details are given retarding how the query is processed. The query processor in claim 1 "determines if any of the plurality of documents match the query by processing the query and comparing the processed query to the plurality of words in the at least one word index, thereby providing a query result." All we know from Johnson is there are a set of case instance descriptions, and that queries may be run on the case instance descriptions. Nowhere does Johnson teach or suggest words in at least one word index. Nowhere does Johnson teach or suggest processing the query and comparing the processed query to the plurality of words in the at least one word index. For these reasons, claim 1 is allowable over Johnson.

Applicants respectfully assert that the examiner has not shown that Johnson teaches each and every limitation in claim 1. As a result, the examiner has failed to establish a prima facie case of anticipation for claim 1 under 35 U.S.C. §102(e). Johnson fails to teach or suggest the load document processor, plurality of documents, word index, and comparing a processed query to the plurality of words in the at least one word index, as recited in claim 1. As a result, claim 1 is allowable over Johnson.

6

Claims 8-10 and 17 are grouped with claim 1, and stand or fall based on the allowability of representative claim 1 discussed above. For the reasons given above, claims 1, 8-10 and 17 are allowable over Johnson, and applicant respectfully requests that the examiner's rejection of claims 1, 8-10 and 17 under 35 U.S.C. §102(e) be reversed.

Claims 2, 11 and 18

In rejecting claim 2, the examiner states that Johnson teaches "wherein the index processor creates at least one word index in response to a build index request from a user", citing col. 64:29-34, col. 51:30-65 and col. 16:50-67. There are two problems with the examiner's rejection of claim 2. First, the indexes in Johnson are not word indexes. The indexes in Johnson are indexes of case instance objects. Second, there is no teaching in Johnson regarding how the indexes are built. The framework in Johnson permits a user to provide an index definition class of objects. By providing the index definition class of objects, the user essentially enables indexes to be used in Johnson. However, providing the index definition class of objects does not amount to a "build index request" from a user that causes the index processor to create at least one word index as recited in claim 2. For these reasons, claim 2 is allowable over Johnson. Claims 11 and 18 are grouped with claim 2, and stand or fall based on the allowability of representative claim 2 discussed above.

It is interesting to note that claim 2 provides additional limitations for the index processor in claim 1, yet the examiner uses the same citations from Johnson that were used to reject the index processor in claim 1. The presence of an index does not imply that the index was created in response to a build index request from a user. For the reasons given above, claims 2, 11 and 18 are allowable over Johnson, and applicant respectfully requests that the examiner's rejection of claims 2, 11 and 18 under 35 U.S.C. §102(e) be reversed.

7

Claims 3, 12 and 19

In rejecting claim 3, the examiner states that Johnson teaches a frequency counter that indicates the number of times a word appears in the at least one word index, citing the same portions of Johnson that were used to reject the index processor in claim 1. The examiner has not identified any teaching in Johnson that reads on the frequency counter in claim 3. As a result, the examiner has failed to establish a prima facie case of anticipation for claim 3 under 35 U.S.C. §102(e). Johnson does not teach word indexes, so Johnson cannot teach a frequency counter as recited in claim 3 that indicates the number of times a *word* appears in the at least one *word index*. Because Johnson does not teach or suggest the frequency counter in claim 3, claim 3 is allowable over Johnson.

It is interesting to note that claim 3 provides additional limitations relating to the number of times a word appears in at least one word index, yet the examiner uses the same citations from Johnson that were used to reject the index processor in claim 1. The presence of an index does not imply the frequency counter recited in claim 3. Claims 12 and 19 are grouped with claim 3, and stand or fall based on the allowability of representative claim 3 discussed above. Applicant respectfully requests that the examiner's rejection of claims 3, 12 and 19 under 35 U.S.C. §102(e) be reversed.

Claims 4, 13 and 20

In rejecting claim 4, the examiner states that Johnson teaches a table that maps a word index to the indexed document from which it was preprocessed, citing the same portions of Johnson that were used to reject the index processor in claim 1. The examiner has not identified any teaching in Johnson that reads on the table in claim 4. As a result, the examiner has failed to establish a prima facie case of anticipation for claim 4 under 35 U.S.C. §102(e). Johnson does not teach word indexes, and does not teach preprocessing a document, so Johnson cannot teach a table as recited in claim 4 that maps a word index

8

to the indexed document from which it was preprocessed. Because Johnson does not teach or suggest the table in claim 4, claim 4 is allowable over Johnson.

It is interesting to note that claim 4 provides additional limitations relating to a table that maps a word index to the indexed document from which it was preprocessed, yet the examiner uses the same citations from Johnson that were used to reject the index processor in claim 1. The presence of an index does not imply the table recited in claim 4. Claims 13 and 20 are grouped with claim 4, and stand or fall based on the allowability of representative claim 4 discussed above. Applicant respectfully requests that the examiner's rejection of claims 4, 13 and 20 under 35 U.S.C. §102(e) be reversed.

Claims 5, 14 and 21

In rejecting claim 5, the examiner states that Johnson teaches the parsing method that identifies text words from other text characters, citing FIGS. 8 & 36 of Johnson and associated text. FIG. 36 of Johnson shows a ParseDefinition object that parses the CaseDefinition object. Note, however, that the parsing method in claim 5 "identifies text words from other text characters." Although Johnson does teach a parser, the parser in Johnson does not identify text words from other text characters. For this reason, claim 5 is allowable over Johnson. Claims 14 and 21 are grouped with claim 5, and stand or fall based on the allowability of representative claim 5 discussed above. Applicant respectfully requests that the examiner's rejection of claims 5, 14 and 21 under 35 U.S.C. §102(e) be reversed.

Claims 6, 15 and 22

In rejecting claim 6, the examiner states that Johnson teaches a stoplist method that 1) identifies text words not containing sufficient information to be useful in providing a query result and 2) deletes such text words, citing col. 66:14-61. The

examiner has not identified any teaching in the cited portion of Johnson that reads on the stoplist method in claim 6. As a result, the examiner has failed to establish a prima facie case of anticipation for claim 6 under 35 U.S.C. §102(e). Johnson does not teach preprocessing of documents, text words, and deleting text words if certain conditions are met. Because Johnson does not teach or suggest the stoplist method in claim 6, claim 6 is allowable over Johnson. Claims 15 and 22 are grouped with claim 6, and stand or fall based on the allowability of representative claim 6 discussed above. Applicant respectfully requests that the examiner's rejection of claims 6, 15 and 22 under 35 U.S.C. §102(e) be reversed.

Claims 7, 16 and 23

In rejecting claim 7, the examiner states that Johnson teaches a stemming method that 1) identifies text word stems of which a text word is formative, and 2) replaces the text word with the stem, citing col. 66:14-61. The examiner has not identified any teaching in the cited portion of Johnson that reads on the stemming method in claim 7. As a result, the examiner has failed to establish a prima facie case of anticipation for claim 7 under 35 U.S.C. §102(e). Johnson does not teach preprocessing of documents, text words, text stems, and replacing a text word with a text stem. Because Johnson does not teach or suggest the stemming method in claim 7, claim 7 is allowable over Johnson. Claims 16 and 23 are grouped with claim 7, and stand or fall based on the allowability of representative claim 7 discussed above. Applicant respectfully requests that the examiner's rejection of claims 7, 16 and 23 under 35 U.S.C. §102(e) be reversed.

CONCLUSION

Claims 1-23 are addressed in this Appeal. For the numerous reasons articulated above, applicants maintain that the rejection of claims 1-23 under 35 U.S.C. § 102(e) is erroneous.

Applicants respectfully submit that this Appeal Brief fully responds to, and successfully contravenes, the sole ground of rejection and respectfully requests that the final rejection be reversed and that all claims in the subject patent application be found allowable.

Respectfully submitted,

Derek P. Martin
Reg. No. 36,595

MARTIN & ASSOCIATES, L.L.C.
P.O. Box 548
Carthage, MO 64836-0548
(417) 358-4700
Fax (417) 358-5757

CLAIMS APPENDIX

1 1.   An apparatus comprising:

2   at least one processor;

3   a memory coupled to the at least one processor;

4   a user-extensible object oriented framework residing in the memory, the

5 framework including at least one core function that cannot be modified by a user and at

6 least one extensible function defined by a user to customize the framework and thereby

7 define a desired information retrieval system, the framework including:

8   a load document processor that loads and preprocesses a plurality of

9   documents;

10   an index processor that creates at least one word index corresponding to

11   the plurality of documents; and

12   a query processor that receives a query and determines if any of the

13   plurality of documents match the query by processing the query and comparing

14   the processed query to the plurality of words in the at least one word index,

15   thereby providing a query result.

1 2.   The apparatus of claim 1 wherein the index processor creates at least one word

2 index in response to a build index request from a user.

1 3.   The apparatus of claim 1 wherein the framework further includes:

2   a frequency counter that indicates the number of times a word appears in the at

3 least one word index.

1 4. The apparatus of claim 1 wherein the framework further includes:

2 a table that maps a word index to the indexed document from which it was

3 preprocessed.

1 5. The apparatus of claim 1 wherein the preprocessing by the load document

2 processor includes a parsing method that identifies text words from other text characters.

1 6. The apparatus of claim 1 wherein the preprocessing by the load document

2 processor includes a stoplist method that 1) identifies text words not containing sufficient

3 information to be useful in providing a query result and 2) deletes such text words.

1 7. The apparatus of claim 1 wherein the preprocessing by the load document

2 processor includes a stemming method that 1) identifies text word stems of which a text

3 word is a formative, and 2) replaces the text word with the stem.

1   8.   A program product comprising:

2   (A) a user-extensible object oriented framework mechanism comprising:

3   (1) a load document processor that loads and preprocesses a plurality of

4   documents;

5   (2) an index processor that creates at least one word index corresponding

6   to the plurality of documents; and

7   (3) a query processor that receives a query and determines if any of the

8   plurality of documents match the query by processing the query and comparing

9   the processed query to the plurality of words in the at least one word index,

10   thereby providing a query result; and

11   (B) computer-readable signal bearing media bearing the framework mechanism.

1   9.   The program product of claim 8 wherein the computer-readable signal bearing

2   media comprises recordable media.

1   10.   The program product of claim 8 wherein the computer-readable signal bearing

2   media comprises transmission media.

1   11.   The program product of claim 8 wherein the index processor creates at least one

2   word index in response to a build index request from a user.

1   12.   The program product of claim 8 wherein the framework mechanism further

2   includes:

3   a frequency counter that indicates the number of times a word appears in the at

4   least one word index.

1   13.   The program product of claim 8 wherein the framework mechanism further

2   includes:

3   a table that maps a word index to the indexed document from which it was

4   preprocessed.

1   14.   The program product of claim 8 wherein the preprocessing by the load document

2   processor includes a parsing method that identifies text words from other text characters.

1   15.   The program product of claim 8 wherein the preprocessing by the load document

2   processor includes a stoplist method that 1) identifies text words not containing sufficient

3   information to be useful in providing a query result and 2) deletes such text words.

1   16.   The program product of claim 8 wherein the preprocessing by the load document

2   processor includes a stemming method that 1) identifies text word stems of which a text

3   word is a formative, and 2) replaces the text word with the stem.

1   17.   A method of retrieving information from a plurality of documents comprising the

2   steps of:

3   (1) providing a user-extensible object oriented framework mechanism;

4   (2) extending the object oriented framework mechanism; and

5   (3) executing the extended object oriented framework mechanism, the executing

6   framework mechanism performing the steps of:

7   (A)   loading and preprocessing a plurality of documents;

8   (B)   creating at least one word index corresponding to the plurality of

9   documents; and

10   (C)   receiving a query and determining if any of the plurality of

11   documents match the query by processing the query and comparing

12   the processed query to the plurality of words in the at least one

13   word index, thereby providing a query result.

1   18.   The method of claim 17 wherein the framework mechanism performs step (B) in

2   response to a build index request from a user.

1   19.   The method of claim 17 wherein the executing framework mechanism further

2   preforms the step of counting the number of times a word appears in the at least one word

3   index.

1   20.   The method of claim 17 wherein the executing framework mechanism further

2   preforms the step of mapping a word index to the indexed document from which it was

3   preprocessed.

1   21.   The method of claim 17 wherein the preprocessing of a document includes the

2   step of identifying text words from other text characters.

1 22. The method of claim 17 wherein the preprocessing of a document includes the

2 steps of:

3   1) identifying text words not containing sufficient information to be useful in

4 providing a query result; and

5   2) deleting such text words.

1 23. The method of claim 17 wherein the preprocessing of a document includes the

2 steps of:

3   1) identifying text word stems of which a text word is a formative; and

4   2) replacing the text word with the stem.